Name : Deep Pawar
CWID : A20545137
Sub : CS536 SOP Assignment 5

Date
Page

1. For formal proof in Assignment 4, Question 10 :

a) Create a corresponding full proof outline under partial correctness.

→ Full proof outline under partial correctness :

$\{n > 0\}$ $k := n - 1$; $\{n > 0 \wedge k = n - 1\}$ $x := n$;
$\{n > 0 \wedge k = n - 1 \wedge x = n\}$

$\{inv\ p \equiv 1 \leq k \leq n \wedge x = n! \div k!\}$
while $k > 1$ do
$\{p \wedge k > 1\}$
$\{p[x * k / x][k - 1 / k]\}$ $k := k - 1$;
$x := x * k$ $\{p\}$
od
$\{p \wedge k \leq 1\}$ $\{x = n!\}$

b) Minimal proof outline under partial correctness :

$\{n > 0\}$ $k := n - 1$; $x := n$;
$\{int\ p \equiv 1 \leq k \leq n \wedge x = n! \div k!\}$

while $k > 1$ do
$k := k - 1$; $x := x * k$
od

$\{x = n!\}$

Q. 2.

→ Given,

$\{ n \geq 0 \}$

$k := 0 ; \quad s := 0 ;$

$\{ \text{inv } p \equiv 0 \leq k \leq n \wedge s = sum(0, k) \}$

while $k < n$ do

$\quad s := s + k + 1 ; \quad k := k + 1$

od

$\{ s = sum(0, n) \}$

We need to calculate full proof of outline under partial correctness using backward assignment for assignments before loop & the forward assignment in the loop body :

$\{n \geq 0\}$ $\{0 \leq 0 \leq n \wedge 0 = sum(0, 0)\}$

$k = 0;$

$\{0 \leq k \leq n \wedge 0 = sum(0, k)\}$

$s = 0;$

$\{inv\ p \equiv 0 \leq k \leq n \wedge s = sum(0, k)\}$

while $k < n$ do

$\{p \wedge k < n\}$

$s := s + k + 1;$

$\{(p \wedge k < n)[s_0/s] \wedge s = (s + k + 1)[s_0/s]\}$

$k := k + 1$

$\{(p \wedge k < n)[s_0/s] \wedge s = (s + k + 1)[s_0/s] \wedge k$
$= (k + 1)[k_0/k]\}$

$\{p\}$

od

$\{p \wedge k \geq n\}$ $\{s = sum(0, n)\}$

Date
Page

Q. 3.

→ Given,

$\{y \geq 1\}\ x := 0;\ r := 1;$

$\{inv\ 1 \leq r = 2^x \leq y\}$

while $2 * r \leq y$ do $r := 2 * r;\ x := x + 1$ od

$\{r = 2^x \leq y \leq 2^{(x+1)}\}$

Full proof outlier under partial correctness:

$\{y \geq 1\}\ x := 0\ \{y = y_0 \wedge y_0 \geq 1 \wedge x = 0\}$

$r := 1;$

$\{y = y_0 \wedge y_0 \geq 1 \wedge x = 0 \wedge r = 1\}$

$\{inv\ 1 \leq r = 2^x \leq y\}$

while $2 * r \leq y$ do

$\{1 \leq r = 2^x \leq y \wedge 2 * r \leq y\}\ r := 2 * r;$

$\{1 \leq r_0 = 2^x \leq y/2 \wedge r = 2 * r_0 = 2^{(x+1)} \leq y\}$

$x := x + 1$

$\{1 \leq r = 2^x \leq y\}$

od

$\{1 \leq r = 2^x \leq y \wedge 2 * r > y\}$

$\{r = 2^x \leq y \leq 2^{(x+1)}\}$

Explanation of logical implication using Forward Assignment Axiom:

① $\{y \geq 1\}\ x := 0;\ \{y = y_0 \wedge y_0 \geq 1 \wedge x = 0\}$

Follows Forward Assignment Axiom, we preserve the initial value of y as y_0, and $x = 0$ to reflect assignment

② $\{y = y_0 \wedge y_0 \geq 1 \wedge x = 0\}$ $r := 1$; $\{y = y_0 \wedge y_0 \geq 1 \wedge$
$x = 0 \wedge r = 1\}$

Again using Forward Assignment Axiom, we preserve the previous conditions & add $r = 1$.

③ In the loop:

$\{1 \leq r = 2^x \leq y \wedge 2 * r \leq y\}$ $r = 2 * r$;
$\{1 \leq r_0' = 2^x \leq y/2 \wedge r = 2 * r_0 = 2^{(x+1)} \leq y\}$

Here, r_0 represents value of r before assignment. We use Forward Assignment Axiom to change r to r_0.

④ $\{1 \leq r_0 = 2^x \leq y/2 \wedge r = 2 * r_0 = 2^{(x+1)} \leq y\}$
$x := x + 1$ $\{1 \leq r = 2^x \leq y\}$

Using Forward Assignment Axiom, we update x to $x+1$ which maintains loop invariants.

Q. 4.

→ Full proof outline under total correctness for Q. 3 :

Bound expression : $y - r$

$\{y \geq 1\}$ $x := 0$; $\{y \geq 1 \land x = 0\}$ $r := 1$;

$\{y \geq 1 \land x = 0 \land r = 1\}$

$\{\text{inv } 1 \leq r = 2^x \leq y \land \text{bd } y - r \geq 0\}$

while $2 * r \leq y$ do

$\{1 \leq r = 2^x \leq y \land 2 * r \leq y \land y - r \geq 0\}$

$r := 2 * r$;

$\{1 \leq 2^x \leq y/2 \land r = 2^{(x+1)} \leq y \land y - r/2 \geq 0\}$

$x := x + 1$

$\{1 \leq r = 2^x \leq y \land y - r \geq 0\}$

od

$\{1 \leq r = 2^x \leq y \land 2 * r > y \land y - r \geq 0\}$

$\{r = 2^x \leq y < 2^{(x+1)}\}$

Explanation of logical implications :

① $\{y \geq 1\}$ $x := 0$; $\{y \geq 1 \land x = 0\}$

Backward Assignment : $(y \geq 1)[0/x] \equiv y \geq 1$

② $\{y \geq 1 \land x = 0\}$ $r := 1$; $\{y \geq 1 \land x = 0 \land r = 1\}$

Backward Assignment :

$(y \geq 1 \land x = 0 \land r = 1)[1/r] \equiv y \geq 1 \land x = 0$

Date
Page

③ $\{y \geq 1 \wedge x = 0 \wedge r = 1\} \Rightarrow \{1 \leq r = 2^x \leq y \wedge bd\ y - r \geq 0\}$

Logical Implication : $1 \leq r$ since $r = 1$, $r = 2^x$ since $1 = 2^0$, $r \leq y$ since $1 \leq y$ (precondition)

④ $\{1 \leq r = 2^x \leq y \wedge 2 * r \leq y \wedge y - r \geq 0\}$ $r := 2 * r$;
$\{1 \leq 2^x \leq y/2 \wedge r = 2^{(x+1)} \leq y \wedge y - r/2 \geq 0\}$

Backward Assignment :

$(1 \leq 2^x \leq y/2 \wedge r = 2^{(x+1)} \leq y \wedge y - r/2 \geq 0)[2 * r / r]$

⑤ $\{1 \leq 2^x \leq y/2 \wedge r = 2^{(x+1)} \leq y \wedge y - r/2 \geq 0\}$ $x := x + 1$;
$\{1 \leq r = 2^x \leq y \wedge y - r \geq 0\}$

Backward Assignment :

$(1 \leq r = 2^x \leq y \wedge y - r \geq 0)[x - 1 / x]$

⑥ $\{1 \leq r = 2^x \leq y \wedge 2 * r > y \wedge y - r \geq 0\} \Rightarrow$
$\{r = 2^x \leq y < 2^{(x+1)}\}$

Logical Implication : $r = 2^x \leq y$ (from first part), $y < 2 * r = 2^x = 2^{(x+1)}$, so $y < 2^{(x+1)}$

Q. 5.

→ Given,

$\{p\}$ if sqrt$(x) > y$ then $x := b[x - y]$ else $y := b[y - x]$ fi $\{x = y\}$

Precondition P :

$P \equiv x \geq 0 \wedge y \geq 0 \wedge \forall i\ (0 \leq i < N \rightarrow b[i] \geq 0)$
$\wedge\ |x - y| < N$

where,

N is the size of array b

Full proof of outline :

$\{ x \geq 0 \wedge y \geq 0 \wedge \forall i . (0 \leq i < N \rightarrow b[i] \geq 0)$
$\wedge \; |x - y| < N \}$

if sqrt (x) > y then

$\{ x \geq 0 \wedge y \geq 0 \wedge \forall i \; (0 \leq i < N \rightarrow b[i] \geq 0)$
$\wedge \; |x - y| < N \wedge sqrt(x) > y \}$

$x := b[x - y]$

$\{ x \geq 0 \wedge y \geq 0 \wedge |x - y| < N \wedge x = b[x_0 - y]$
$\wedge \; sqrt(x_0) > y \}$

else

$\{ x \geq 0 \wedge y \geq 0 \wedge \forall i . (0 \leq i < N \rightarrow b[i] \geq 0)$
$\wedge \; |x - y| < N \wedge sqrt(x) \leq y \}$

$y := b[y - x]$

$\{ x \geq 0 \wedge y \geq 0 \wedge |x - y| < N \wedge y = b[y_0 - x]$
$\wedge \; sqrt(x) \leq y_0 \}$

fi

$\{ x = y \}$

Here, y_0 is the original value of y before the assignment

Q. 6.

→ Given,

$\{ sqrt(x) \leq y \}$ $x := x * y;$ $x := 1 \div x$ $\{q\}$

Full proof of outline:

$\{ x \geq 0 \wedge sqrt(x) \leq y \wedge x * y \neq 0 \}$
$x := x * y;$
$\{ x > 0 \wedge x = x_0 * y \wedge sqrt(x_0) \leq y \}$
$x := 1 \div x;$
$\{ x = 1 \div (x_0 * y) \wedge x > 0 \wedge x \leq 1/y \}$

Here,
x_0 is the initial value of x.

Date
Page

Q. 7.

a) Let $\sigma \models p$, then $\perp_d \notin M(W, \sigma)$

→ True

Explanation: If p is an invariant, then it ensures specific constraints on the ~~top~~ loop's execution, preventing the loop from entering states where bd is violed or program does not terminate properly. Thus convergence is guaranteed by the existence of bound expression.

b) The value of t can be negative after the execution of the last iteration of W.

→ False

Explanation: If t is used as a bound expression for the loop W, it must remain non-negative throughout the loop's execution to guarantee proper termination.

c) $sp(p \wedge B \wedge t = t_0, s) \Rightarrow t < t_0$

→ True

Explanation: If $sp(p \wedge B \wedge t = t_0, s)$ holds, meaning the program is in the ~~same~~ state where $t = t_0$ at the start of the current iteration, then after executing the body of the loop, t must be less than t_0 due to the decreasing nature of the bound variable t.

d) $p \wedge t > 0 \Rightarrow B$

→ False

Explanation: The fact that $t > 0$ does not necessarily imply that B holds & thus does not guarantee another iteration. The loop could terminate due to reasons other than t reaching zero.

Date
Page

e) $t < 0 \Rightarrow \neg p$

→ True

Explanation: It is contra-positive of $p \Rightarrow t \geq 0$, which guaranteed by the definition of bound expression. Since t is a bound expression it should never be negative thus $t < 0$ is False & False implies anything is True.

Q. 8.

a) $x - k + n$

→ No

Explanation: without evidence that $x - k + n \geq 0$ at the beginning, we cannot consider $x - k + n$ is a valid bound function for W.

Q. 8.

b] n - k

→ Yes

Explanation: n - k satisfies all the conditions of a bound function for the loop W, as it starts non negative, decreases with each iteration, & reaches zero or negative upon loop termination.

c) $n - k + c$

→ Yes

Explanation: Since $n-k+c$ is non-negative initially, decreases by a fixed amount with each iteration, & will eventually reach zero, $n - k + c$ can be a valid bound function for w.

d) $-k - c$

→ No

Explanation: Because $k - c$ increase in each iteration rather than decreasing, it cannot be used as a bound function for w.

e) $2^{n} \cdot 2^{c-k}$

→ Yes

Explanation : Since $2^{n} \cdot 2^{c-k} = 2^{n+c-k}$ is positive initially, decreases by a factor of 2 in each iteration, & approaches zero as the loop progresses, it can be considered a valid bound function for W.

Q. 9.

→ Below are the 5 possible candidates for the loop invariant p & their corresponding loop condition B:
Here, we are using u as the fresh variable:

① $p_1 \equiv y \geq u \wedge x = 2 * y \leq n < 3 * (y+1)$, and $B_1 \equiv u \neq 0$

② $p_2 \equiv y \geq 0 \wedge x = u * y \leq n < 3 * (y+1)$, and $B_2 \equiv u \neq 2$

③ $p_3 \equiv y \geq 0 \wedge x = 2 * y \leq u < 3 * (y+1)$, and $B_3 \equiv u \neq n$

④ $p_4 \equiv y \geq 0 \wedge x = 2 * y \leq n < u * (y+1)$, and $B_4 \equiv u \neq 3$

⑤ $p_5 \equiv y \geq 0 \wedge x = 2 * y \leq n < 3 * (y+u)$, and $B_5 \equiv u \neq 1$

Q. 10.

→ Below are the 4 possible candidates for the loop invariant p & their corresponding loop condition B:

① $p_1 \equiv (z = 2^{y}) \wedge (2^{y} \leq x) \wedge (x < 2^{y+1})$, and $B_1 \equiv y < 0$.

② $P_2 \equiv (y \geq 0) \wedge (2^y \leq x) \wedge (x < 2^{y+1})$, and
$B_2 \equiv z \neq 2^y$

③ $P_3 \equiv (y \geq 0) \wedge (z = 2^y) \wedge (x < 2^{y+1})$, and
$B_3 \equiv 2^y > x$

④ $P_4 \equiv (y \geq 0) \wedge (z = 2^y) \wedge (2^y \leq x)$, and
$B_4 \equiv x \ngeq 2^{y+1}$